AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A document-generation process performed using a computer system, the process comprising:

parsing a raw document to create an internal representation of the document;

reading a first first-level transform from a transform database;

applying the first first-level transform to the internal representation ~~so as~~ to create a first first-level document;

writing the first first-level document to memory;

reading a second first-level transform from the transform database;

applying the second first-level transform to the internal representation to create a second first-level document;

receiving a first request for a second-level document that depends from the first first-level document;

in response to the first request, reading a second-level transform from the transform database;

in response to the first request, applying the second-level transform to the first first-level document ~~so as~~ to create [[a]] the second-level document; ~~and~~

writing the second-level document to memory; and

writing the second first-level document to memory;

wherein the first first-level document and the second first-level document are different.

2. (Currently Amended) A document-generation process as defined in Claim 1, further comprising:

revising the raw document;

applying the first first-level transform to the revised raw document ~~so as~~ to create a revised first first-level document;

writing the revised first first-level document to memory; and

indicating that a then-existing second-level document is invalid.

3. (Original) A document-generation process as defined in Claim 2, further comprising:

receiving a second request for the then-existing second-level document;

determining that the then-existing second-level document has been indicated invalid;

applying the second-level transform to the revised first first-level document ~~so as~~ to create a revised second-level document; and

writing the revised second-level document to memory.

4. (Currently Amended) A document-generation process as defined in Claim 1, further comprising:

receiving a request for a revised first first-level transform;

revising the then-existing first first-level transform;

applying the revised first first-level transform to the raw document ~~so as~~ to create a revised first first-level document;

writing the revised first first-level document to memory; and

indicating that the then-existing second-level document is invalid.

5. (Currently Amended) A document-generation process as defined in Claim 4, further comprising:

receiving a second request for the then-existing second-level document;

determining that the then-existing second-level document has been indicated invalid;

applying the second-level transform to the revised first first-level document ~~so as~~ to create a revised second-level document; and

writing the revised second-level document to memory.

6. (Original) A document-generation process as defined in Claim 1, wherein a respective GID is assigned to each of the first-level document and the second-level document.

7. (Currently Amended) A document-generation process as defined in Claim 1, wherein the first first-level document and the second-level document are timestamped and wherein a document is deleted when a timestamp indicates that the document is stale.

8. (Currently Amended) A method of generating customized versions of a document using a computer system, the method comprising:

storing the document in raw form;

parsing the document to create an internal representation of the document;

receiving a request to generate a second-level document; ~~and~~

decomposing the document to create the second-level document, wherein

decomposing the document comprises:

applying a first first-level transform to the internal representation of the document ~~so as~~ to create a first first-level document; [[and,]] and

in response to the request to generate the second-level document, applying a second-level transform to the first first-level document ~~so as~~ to create the second-level document; and

decomposing the document to create a second first-level document, wherein

decomposing the document to create the second first-level document comprises:

applying a second first-level transform to the internal representation of the document to create a second first-level document;

wherein the first first-level document and the second first-level document are different.

9. (Currently Amended) A method as defined in Claim 8, wherein applying a first first-level transform and applying a second-level transform comprises applying sequential transforms to the document.

10. (Canceled)

11. (Canceled)

12. (Currently Amended) A method as defined in Claim 8, wherein decomposing the document comprises applying a third-level transform to the second level document ~~so as~~ to create a third-level document.

13. (Original) A method as defined in Claim 8, wherein the document is stored in raw XML form.

14. (Currently Amended) A method as defined in Claim 13, wherein applying a first first-level transform and applying a second-level transform comprises applying sequential transforms to the document.

15. (Currently Amended) A method as defined in Claim 14, wherein applying a first first-level transform of the document stored in raw XML form comprises applying a subscription-level transform to the internal representation of the document ~~so as~~ to create a subscription-level document.

16. (Previously Presented) A method as defined in Claim 15, wherein the subscription-level transform enables content filtering of the internal representation in accordance with a user's request.

17. (Currently Amended) A method as defined in Claim 16, wherein applying a second-level transform comprises applying an organization-level transform to the subscription-level document ~~so as~~ to create an organization-level document.

18. (Original) A method as defined in Claim 17, wherein the organization-level transform enables subscription-specific content filtering of a subscription-level document.

19. (Currently Amended) A method as defined in Claim 18, wherein decomposing the document comprises applying a presentation-level transform to the organization-level document ~~so as~~ to create a presentation-level document.

20. (Original) A method as defined in Claim 19, wherein the presentation-level transform generates an organization-specific document for end user presentation.

21. (Original) A method as defined in Claim 20, wherein the presentation-level transform generates an HTML document or a text file for end user presentation.

22. (Original) A method as defined in Claim 21, wherein the subscription-level transform is mandatory and the organization-level and presentation-level transforms are optional.

23. (Original) A method as defined in Claim 8, wherein a transform is applied to a document only as a result of an initial demand for a transformed document.

24. (Original) A method as defined in Claim 23, wherein the demand for a transformed document is a client request.

25. (Original) A method as defined in Claim 24, wherein the demand for a transformed document is a document publication process.

26. (Original) A method as defined in Claim 23, wherein transformed documents are written to a cache.

27. (Original) A method as defined in Claim 26, wherein demands for a transformed document, subsequent to the initial demand, are referred to the cache.

28. (Currently Amended) A computer readable medium having data stored therein to cause a data processing system to generate a data document according to a process comprising:

storing a raw form of the document;

parsing the document to create an internal representation of the document; and

receiving a request from a client computer system coupled to the data processing system to generate a second-level document into a particular form;

decomposing the document into the form requested by the client system, wherein decomposing the document comprises:

applying a first first-level transform to the internal representation of the document ~~so as~~ to create a first first-level document; [[and,]] and

in response to the request to generate the second-level document, applying a second-level transform to the first first-level document ~~so as~~ to create the second-level document; and

decomposing the document to create a second first-level document, wherein decomposing the document to create the second first-level document comprises:

applying a second first-level transform to the internal representation of the document to create a second first-level document;

wherein the first first-level document and the second first-level document are different.

29. (Currently Amended) A computer readable medium as defined in Claim 28, wherein applying a first first-level transform and applying a second-level transform comprises applying sequential transforms to the document.

30. (Previously Presented) A computer readable medium as defined in Claim 28, wherein the document is stored in XML form.

31. (Previously Presented) A computer readable medium as defined in Claim 30, wherein the document stored in XML form is parsed by an XML parser to create the internal representation.

32. (Previously Presented) A computer readable medium as defined in Claim 31, wherein the internal representation level of the document is transformed to a subscription-level document by applying a subscription-level transform to the internal representation.

33. (Currently Amended) A computer readable medium as defined in Claim 32, wherein application of the subscription level transform to the internal representation ~~so as~~ to create a subscription-level document is required.

34. (Previously Presented) A computer readable medium as defined in Claim 32, wherein the subscription-level document is transformed into an organization-level document by applying an organization-level transform to the subscription-level document.

35. (Previously Presented) A computer readable medium as defined in Claim 34, wherein application of the organization-level transform to the subscription-level document ~~so as~~ to create an organization-level document is optional.

36. (Previously Presented) A computer readable medium as defined in Claim 34, wherein the internal representation of the document is decomposed to a transform-level document only in response to a request for a transform-level document.

37. (Previously Presented) A computer readable medium as defined in Claim 36, wherein transformed documents are written to a cache.

38. (Previously Presented) A computer readable medium as defined in Claim 37, wherein an initial request for a transformed document causes decomposition of the internal representation into the form requested and wherein subsequent requests for a transformed document causes the transformed document to be retrieved from memory.

39. (Previously Presented) A computer readable medium as defined in Claim 29, wherein the data document is generated according to a process comprising:

tracking the dependencies of a transformed document; and

regenerating the transformed document when any dependency related to the document changes.

40. (Previously Presented) A computer readable medium as defined in Claim 39, wherein the document is generated according to a process comprising:

designating a cached version of the document invalid when any dependency related to the document changes, and

regenerating the transformed document in response to a request for the document that is made after the dependency change.

41. (Previously Presented) A computer readable medium as defined in Claim 40, wherein the document is stored in XML form.

42. (Previously Presented) A computer readable medium as defined in Claim 39, wherein the document stored in XML form is parsed by an XML parser to create the internal representation.

43. (Previously Presented) A computer readable medium as defined in Claim 42, wherein the internal representation level of the document is transformed to a subscription-level document by applying a subscription-level transform to the internal representation.

44. (Previously Presented) A computer readable medium as defined in Claim 43, wherein application of the subscription level transformed to the internal representation ~~so as~~ to create a subscription-level document is required.

45. (Previously Presented) A computer readable medium as defined in Claim 43, wherein the subscription-level document is transformed into an organization-level document by applying an organization-level transform to the subscription-level document.

46. (Previously Presented) A computer readable medium as defined in Claim 45, wherein application of the organization-level transform to the subscription-level document ~~so as~~ to create an organization-level document is optional.

47. (Previously Presented) A computer readable medium as defined in Claim 45, wherein the internal representation of the document is decomposed to a transform-level document only in response to a request for a transform-level document.

48. (Previously Presented) A computer readable medium as defined in Claim 47, wherein transformed documents are written to a cache.

49. (Canceled)

50. (Canceled)

51. (Canceled)

52. (Canceled)

53. (Currently Amended) A method of generating customized versions of a document using a computer system, the method comprising:

storing the document in a primitive form;

transforming the document from primitive form into an internal representation of the document;

transforming the internal representation into at least ~~one subscription-level~~ a first first-level document[[,]] and into a ~~DEFAULT organization-level~~ second first-level document ~~and into at least one user-specific organization-level document~~;

transforming the ~~DEFAULT organization-level~~ first first-level document into at least one ~~presentation-level~~ first second-level document; and

transforming the ~~user-specific organization-level~~ second second-level document into at least one ~~presentation-level~~ second second-level document;

wherein the first first-level document and the second first-level document are different.

54. (Currently Amended) A method of generating customized versions of documents as defined in Claim [[53]] 71, wherein the user-specific organization-level document is transformed into at least two presentation-level documents.

55. (Original) A method of generating customized versions of documents as defined in Claim 54, wherein the user-specific organization-level document is transformed into an HTML presentation-level document and into a FLAT presentation-level transform.

56. (Currently Amended) A method of generating customized versions of a document as defined in Claim [[53]] 71, wherein:

(i) the internal representation is transformed into a first subscription-level document and into a second subscription-level document;

(ii) the first subscription level document is transformed into a subscription-level specific DEFAULT organization-level document and into at least one user-specific organization-level document; and

(iii) the second organization-level document is transformed into a subscription-level-specific DEFAULT organization-level document.

57. (Original) A method of generating customized versions of a document as defined in Claim 56, wherein the document is stored in XML form.

58. (Currently Amended) A system for the generation of customized data documents, the system comprising:

first database for storing raw data documents;

first tabular means for storing document records;

an interface coupling the first database to the first tabular means;

a request interface to receive a document generation request, wherein the document generation request indicates a particular document type;

a second database for storing a compilation of transforms that enable an internal representation of a document to be transformed into a first first-level

document and into a second first-level document and that enable the first first-level document to be transformed into a second-level document, wherein each transform is mapped to a particular document type;

a document generator, coupled to the first and second databases and first and second tabular means, and to the request interface, to generate the first ~~first level~~ first-level document using at least one of the transforms, to generate the second first-level document using at least one of the transforms that is different than at least one of the transforms used to generate the first-level document, and to generate the second-level document using at least one of the transforms in response to receipt of the document generation request, wherein the document type of the second-level document and the transform used to generate the second-level document are indicated by the document generation request;

second tabular means for storing transform records; and

an interface coupling the second database to the second tabular means; and

wherein the first first-level document and the second first-level document are different.

59. (Original) A system as defined in Claim 58, further comprising:

a cache coupled to the first tabular means and to the second tabular means.

60. (Currently Amended) A system as defined in Claim 58, wherein the second database stores a compilation of transforms that enable:

transforming an internal representation into at least one subscription-level document, into a DEFAULT organization-level document and into at least one user-specific organization-level document;

transforming the DEFAULT organization-level document into at least one presentation-level document; and

transforming the user-specific organization-level document into at least one presentation-level document.

61. (Previously Presented) A system as defined in Claim 60, wherein the second database stores a compilation of transforms that enable:

(i) the internal representation to be transformed into a first subscription-level document and into a second subscription-level document;

(ii) the first subscription level document to be transformed into a subscription-level specific DEFAULT organization-level document and into at least one user-specific organization-level document; and

(iii) the second organization-level document to be transformed into a subscription-level-specific DEFAULT organization-level document.

62. (Original) A system as defined in Claim 61, further comprising: a cache coupled to the first tabular means and to the second tabular means.

63. (Previously Presented) A document-generation process as defined in Claim 1 wherein receiving a first request for a second-level document that depends from the first-level document comprises:

receiving a first request from a client computer coupled to the computer system for a second-level document that depends from the first-level document

64. (Currently Amended) A document-generation process as defined in Claim 1 further comprising:

tracking the dependencies of a transformed document, wherein the transformed document includes at least one [[or]] of the first first-level document and the second-level document and the dependencies include the first first-level document and the internal representation; and

regenerating the transformed document when any dependency related to the document changes.

65. (Previously Presented) A method of generating customized versions of a document using a computer system as defined in Claim 8 wherein receiving a request to generate a second-level document comprises:

receiving a request from a client computer coupled to the computer system to generate a second-level document.

66. (Currently Amended) A method of generating customized versions of a document using a computer system as defined in Claim 8 further comprising:

tracking the dependencies of a transformed document, wherein the transformed document includes at least one [[or]] of the first first-level document and the second-level document and the dependencies include the first first-level document and the internal representation; and

regenerating the second-level document when any dependency related to the transformed document changes.

67. (Currently Amended) A method of generating customized versions of a document using a computer system as defined in Claim [[53]] 71 wherein receiving a request to generate a second-level document comprises:

receiving a request from a client computer coupled to the computer system to generate [[a]] the second-level document.

68. (Currently Amended) A method of generating customized versions of a document using a computer system as defined in Claim [[53]] 71 further comprising:

tracking the dependencies of the transformed documents, wherein the transformed documents include the subscription-level document, the DEFAULT organization-level document, the user-specific organization-level document, the presentation-level documents, and the dependencies include the DEFAULT organization-level document, the user-specific organization-level document, and the internal representation; and

regenerating the second-level document when any dependency related to the transformed document changes.

69. (Currently Amended) A method of generating customized versions of a document using a computer system as defined in Claim [[53]] 71 wherein receiving a request to generate a second-level document comprises:

receiving a request from a client computer coupled to the computer system to generate [[a]] the second-level document.

70. (Currently Amended) A system for the generation of customized data documents as defined in Claim 58 further comprising:

a document dependency tracker and regenerator, coupled to the first and second databases and first and second tabular means, to track the dependencies of a transformed document, wherein the transformed document includes at least one or the first-level document and the second-level document and the dependencies include the first-level document and the internal representation and to regenerate the transformed document when any dependency related to the document changes.

71. (New) A method of generating customized versions of documents as in Claim 53, wherein the first-level document comprises a subscription-level document and the second first-level document comprises a DEFAULT organization-level document, wherein:

transforming the internal representation further comprises:

transforming the internal representation into at least one subscription-level document, into a DEFAULT organization-level document and into at least one user-specific organization-level document;

transforming the first first-level document further comprises:

transforming the DEFAULT organization-level document into at least one presentation-level document; and

the method further comprises:

transforming the user-specific organization-level document into at least one presentation-level document.

72. (New) A document-generation process performed using a computer system, the process comprising:

parsing a raw document to create an internal representation of the document;

reading a first first-level transform from a transform database;

applying the first first-level transform to the internal representation to create a first first-level document;

writing the first first-level document to memory;

reading a second first-level transform from the transform database;

applying the second first-level transform to the internal representation to create a second first-level document;

reading a second-level transform from the transform database;

applying the second-level transform to the first first-level document to create a second-level document;

writing the second-level document to memory; and

writing the second first-level document to memory;

wherein the first first-level document and the second first-level document are different.

73. (New) A computer readable medium comprising data stored therein to cause a data processing system to:

read a first first-level transform from a transform database;

apply the first first-level transform to the internal representation to create a first first-level document;

write the first first-level document to memory;

read a second first-level transform from the transform database;

apply the second first-level transform to the internal representation to create a second first-level document;

read a second-level transform from the transform database;

apply the second-level transform to the first first-level document to create a second-level document;

write the second-level document to memory; and

write the second first-level document to memory;

wherein the first first-level document and the second first-level document are different.